From Sierpinski Carpets to Directed Graphs

Hülse, Martin

Published in:
Complex Systems

Publication date:
2010

Citation for published version (APA):
Hülse, M. (2010). From Sierpinski Carpets to Directed Graphs. *Complex Systems*, *19*(1), 45-71.
http://hdl.handle.net/2160/6086

From Sierpiński carpets to directed graphs

Martin Hülse

Computer Science Department, University of Wales, Aberystwyth, Aberystwyth, Wales, SY23 3DB, UK

We introduce a simple method for a deterministic generation of directed and connected graphs. The method is inspired by Sierpiński carpets forming fractal sets. Despite the large size these graphs can have, the distance between most of the nodes is short, i.e. it scales with *log*. We will further show that important network properties, such as degree distribution, can directly be determined by the initial structure of this process. These findings lead us to the formulation of general conditions providing a targeted generation of complex networks for initial structures of arbitrary dimension. Under which circumstances these graphs can show scale-free and small-world properties is discussed as well. As a possible application of this method we will finally discuss the generation of artificial neural networks.

1. Introduction

Artificial neural networks (ANN) represent a method of information processing which is inspired and motivated by the neural structures found in biological systems [15]. Therefore it is not surprising, that ANN are frequently utilized as the basic building blocks for large-scale models in order to explore the nature of complex information processing exploited in animals and human beings.

The majority of such neural models are based on connectivity structures which match with the classical types of ANN, such as, Multilayered-perceptrons, Hopfield-networks or Elman-networks [5, 6]. But, all these network types establish only fully connected networks. The application of fully connected networks, however, might become crucial with respect to plausibility if they are intended to model biological systems. Fully connected ANN can hardly represent brain-like neural structures, if, as only one example, approximately 10^{11} neurons in the human brain are coordinated by "only" 10^{15} synapses.

An alternative, in particular for large-scale neural models, to overcome fully connected neural networks is the creation of random graph structures [3]. Nevertheless, random graph models do not well describe some essential properties of real-world networks [13].

Therefore, we argue, while modeling large-scale neural networks one must consider alternatives for the projections between neural assem-

blies; alternatives which go beyond random graphs and fully connected structures.

Furthermore, it becomes more and more common in the field of Neural Computation [1] that large-scale neural models are used for robot control [7, 17]. This leads very often to implementations of ANN on special hardware devices, like massively parallel processor array VLSI circuits [2]. Such implementations on autonomous robots might be motivated as a proof of concepts as well as for targeting specific issues of embodiment [14]. However, autonomous robots have usually very limited computational resources, especially memory, compared with the performance provided by clusters or similar equipment in research institutions. Hence, for performance reasons it becomes important to utilize highly connected and robust networks established by as less connections as possible.

The objective of this paper is to introduce a deterministic method which enables us to create highly connected and structured neural systems formed by a number of connection magnitudes smaller than needed for fully connected networks. The generation process is inspired by fractal sets. This makes the resulting networks very distinct compared with random graphs and as we will see, they can cover, depending on the initialization, a wide range between fully connected graphs and connected graphs organized as rings. Due to the simplicity and deterministic character of the generation process, this method seems to be a promising alternative for the generation of graphs and opens a wide filed for application in many areas of neural modeling.

Fractal sets, invented and promoted by Mandelbrot [10], are established tools for describing and modeling complex structures and processes, such as textures of surfaces or even the state space of chaotic attractors. Sierpiński carpets are well known examples of mathematical shapes forming fractals (see Figure 1). Inherent properties of fractals, being self-similar and scale-free, can impressively be demonstrated with these sets. Therefore, we have asked, what types of graphs or networks can be expected to emerge, if the intermediate sets, resulting from a generation process towards Sierpiński carpets, are interpreted as adjacency matrixes.

This work presents an investigation of the properties of graphs constructed in the same fashion as Sierpiński carpets. As we will show, the result of this investigation is a simple method for a deterministic generation of *directed and connected graphs*. The manifold of possible graphs provided by this method is systematically analyzed for a low dimensional case. But as we will see, this analysis already leads us to general conditions which guaranty robustness as well as specific degree-distributions for arbitrary dimensions.

Based on these finding, we will introduce two strategies which allow an application of our graph generation method in the domain of

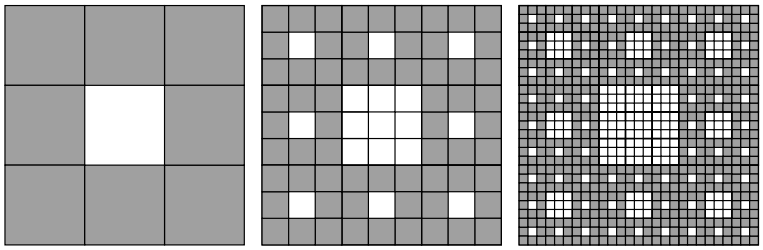


Figure 1. An example for a Sierpiński carpet. Its mask and initial form (left) as well as the first (middle) and second (right) iteration of the process generating the Sierpiński carpet.

artificial neural networks. But before we will discuss these aspects in detail, the next chapter introduces basic definitions and explains the process of graph generation. This includes the formulation of one sufficient condition maintaining connectedness in general. After this, the next two chapters describe essential network properties (shortest paths, degree distribution, clustering, robustness etc.) resulting from a systematic analysis of a representative subset. This is followed by a discussion summarizing our findings in a more general form leading to the outline of possible applications for the generation of artificial neural networks, as aforementioned.

2. The generation of graphs of fractal dimensions

A graph G is a set of vertices V (sometime also called nodes) and connections E (*edges*) between them. An edge e_{ji} connects always only two nodes v_i and v_j, where e_{ji} is the in-coming edge for v_j and the out-going for v_i.

A graph is called directed if there exists at least one pair of nodes v_i and v_j with $e_{ji} \in E$, but $e_{ij} \notin E$ for $i \neq j$. An undirected graph can be represented by a directed graph having two edges between the connected nodes, each in one direction.

We call the number of in-coming and out-going edges the degree k. In an undirected graph the number of in-coming edges k_i is equal to the number of the out-going k_o.

If there exists a path between each pair of nodes in the graph, we call it connected. Notice, in a directed graph it might be the case that node v_j can be reached from v_i, while there is no connection starting in v_j. In this case, the graph is not connected.

The structure of a graph $G(V, E)$ can be represented by an adjacency matrix M. Each matrix element m_{ji} of an adjacency matrix can either be zero or one. The element m_{ji} is one, if and only if $e_{ji} \in E$. In

consequence, the adjacency matrix is symmetric for undirected graphs.

As we will see in the following, adjacency matrixes build the bridge between Sierpiński carpets and graphs as well as they give us a process for the deterministic development of directed and connected graphs.

In Figure 1 an example for the construction of a Sierpiński carpet is illustrated. The process starts with a basic form, a square. Each side of this square is segmented into three equal sections defining an overall partition into 9 identical squares. Some of this squares are labeled, indicated by the grey coloring. We also call this basic form a *mask*.

As one can see in the figure, the Sierpiński carpet is easily generated by applying the same partitioning, which is defined by the mask, for all the labeled squares of the given form. But in contrast to the original generation of Sierpiński carpet, now also unlabeled squares are subdivided in the same way. The only difference is that resulting sub-squares remains unmarked too.

The partitioning of labeled and unlabeled squares including new labeling of the new set of squares is what we call an *iteration*.

Speaking precisely, a distinction has to be made between a mask and the form which the mask is applied to. A mask can be applied to any given form divided into arbitrary number of squares. In the following we use always the mask also as initial form. All the resulting forms therefore are determined by the mask and the number of iterations.

For infinity iterations we get the Sierpiński carpet, that is a set of fractal dimension.

We utilize this type of fractal generating process in order to generate directed and connected graphs simply by interpreting the resulting sets after n iterations as an adjacency matrixes of a graph. Examples of 3-segmented forms transformed into graphs are given in Figure 2. The labeled squares are interpreted as edges, i.e. gray colour represents value 1 in the corresponding adjacency matrix, while unlabeled squares indicate the zero entries.

Given this simple idea we now ask what kind of graphs can be generated by such a process. However, the resulting graphs are obviously fully determined by the mask and the number of iterations. Therefore, our focus for this investigation is directed to the masks. For the three segmented case there exist $2^9 = 512$ different masks. Higher segmentations S generate $2^{(S \cdot S)}$ different masks, since an adjacency matrix has to have equal dimensions.

In order to cope with this exponentially increasing number of possible graphs, we begin our analysis for $S = 3$. This analysis will provide us with insights about the interrelation between mask properties and global structure of the resulting graphs and therefore, will guide us in the huge increasing space of graphs spanned by the masks of higher segmentations.

A first constraint for our investigation is that we only consider con-

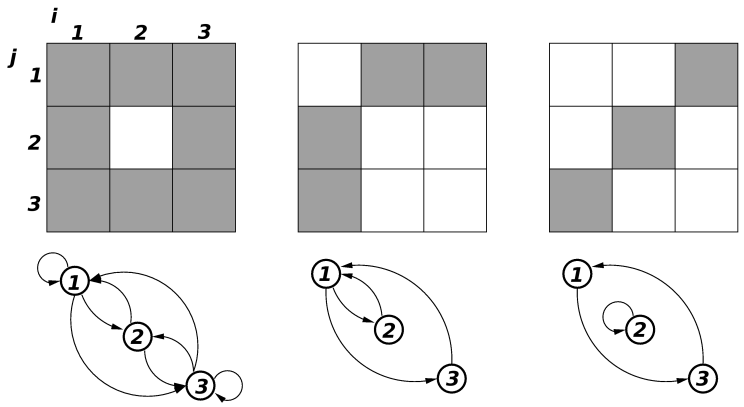


Figure 2. Examples of graphs derived from 3-segmented forms.

nected graphs. This reduces already the number of masks. We will further see, that this set can even be more reduced by taking into account certain symmetries of the adjacency matrixes, which maintain the connectedness of the corresponding graphs. In that way we get a manageable number of remaining basic forms for segmentation 3.

2.1 Labeling and filtering of the masks

The first thing we start with is introducing a general numbering or labeling of the masks. As an unique numbering we have chosen the binary code directly derived from the structure of the mask (see Figure 3 (left)). One can see, that the labeled and unlabeled mask elements are interpreted as 1 and 0 of a binary number respectively. However, this numbering is only unique if the mask segmentation is taking into account. Therefore, we use the symbol M_n^S, where n is the number, which binary representation corresponds to the mask of segmentation S. An example is given in Figure 3 (right). One can see that the number M_{511}^3 refers to the 3-segmented mask in which all segments or entries are labeled. In the 4-segmented case number 511 represents a totally different mask structure.

As we already mentioned above we only consider connected graphs. A necessary condition for a connected graph is that each node has at least one out-going and one in-coming edge, $k_{o/i} > 0$. These values can directly derived from the corresponding adjacency matrix M. Value k_o of node v_l is the sum over all entries in column l, while k_i is the sum over row l:

$$k_o(v_l) := \sum_j m_{j,l}$$

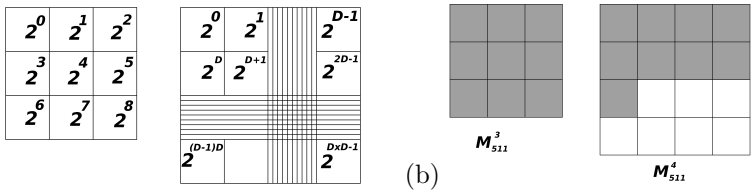


Figure 3. Schema of the 3-dimensional and D-dimensional mask indicating the value of each entry in order to derive the number of the binary code given by the mask (a). Examples of number 511 which binary code represents different mask depending on the dimension of the mask (b).

$$k_i(v_l) := \sum_i m_{l,i}$$

Taking into account the generation process and assuming that the applied mask is also the initial structure, one can see, that each mask containing a node of degree zero generates a graph with at least one node with degree zero. Hence, if a mask of segmentation S generates connected graphs then this mask has at least S labeled entries. In other words, a mask must contain at least as many labeled squares (edges) as columns/rows (nodes).

If the number of edges and nodes are the same, only a ring can create a connected graph. Due to the generation process of such "ring masks" it follows that in the resulting graph the number of nodes is again equal to the number of edges. Hence, the result must be a ring again. Otherwise, the graph wouldn't be connected anymore. But our simulations for the 3-segmented cases have shown that the resulting graphs are not connected anymore. They are forming separated rings and therefore the graph is not connected.

In the following we give an explanation of this phenomenon, which enable us to give a general criteria for the generation of connected graphs.

2.2 Connectedness

Assume a mask M_0 of segmentation S is given. M forms a ring. We further apply a numbering of the nodes which corresponds to the column number in the corresponding adjacency matrix. The ring structure of the graph can be represented by the sequence of nodes while travelling along the path formed by the ring. Without losing generality, we assume this sequence is:

$$(1) \to (2) \to \cdots \to (S-1) \to (S) \to (1) \to \ldots$$

After the first iteration, we get a new adjacency matrix M_1 of dimension (S^2, S^2) representing S^2 nodes connected by S^2 edges.

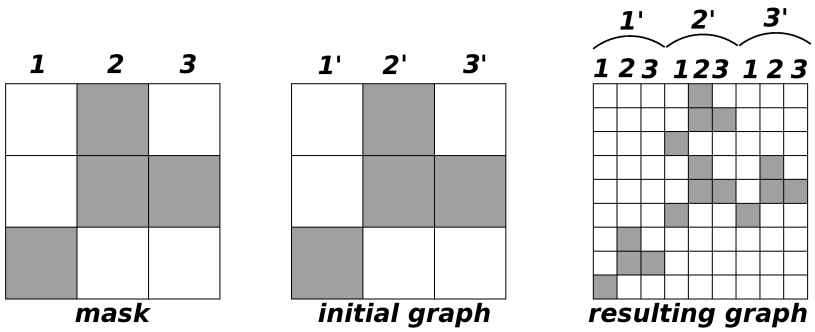


Figure 4. An example for the renumbering after a mask is applied to a graph.

At this point we apply a new numbering system for the nodes represented by M_1. A node v_i in M_1 is symbolized by a pair (k', l), $1 \leq k, l \leq S$, with:

$$k' = 1 + ((i-1) \div S),$$

$$l = 1 + ((i-1) \mod S),$$

where i ($1 \leq i \leq S^2$) is the column number of the adjacency matrix. (See Figure 4.) Due to the construction process, we know, that the resulting graph M_1 only contains nodes with one in-coming and one out-going connection. There is no node with a self-connection in M_1.

Assume, we select a node and write down the sequence of nodes, formed by the path staring in this node. Obviously we can expect to get a sequence like:

$$(k'_1, l_1) \to (k'_2, l_2) \to \cdots \to (k'_n, l_n) \to (k'_1, l_1)$$

If $n = S^2$ we have a ring. But our observation tells us that $n < S^2$. In fact, we have different disjunctive sequences representing the separated rings in M_1. But, how can we derive the resulting sequences? To solve this problem, one has to understand that k' represents the "macro"-connectivity structure of the graph, while l is representing "micro"-connectivity structure. The latter is given by the mask, which turns a given adjacency matrix into a new, larger adjacency matrix. The first, the "macro"-structure, is given by the adjacency matrix, which the mask is applied to. In our case macro- and micro-structure are the same, (since we applied the mask to itself).

In consequence the sequence of $l_1, l_2, \ldots$ must follow the same sequence of the micro-structure, given by the mask. The same holds for the sequence of the k' determined by the macro-structure, the original graph.

Assume we start with node $(3,7)$ with respect to the macro-structure we get the following sequence:

$$(3,7) \rightarrow (4,l_2) \rightarrow (5,l_3) \rightarrow \ldots$$

$$\cdots \rightarrow (S-1,l_u) \rightarrow (S,l_{u+1}) \rightarrow (1,l_{u+2}) \rightarrow \ldots$$

$$\cdots \rightarrow (2,l_w) \rightarrow (3,l_{w+1}) \rightarrow (4,l_{w+2}) \rightarrow \ldots$$

Developing this sequence further with respect to the micro-structure, the second variable l is replaced in the same way. The resulting sequence is the same given in the original mask:

$$(3,7) \rightarrow (4,8) \rightarrow (5,9) \rightarrow \ldots$$

$$\cdots \rightarrow (S-1,((S-1+4) \mod S)=3)) \rightarrow (S,4) \rightarrow (1,5) \rightarrow \ldots$$

$$\cdots \rightarrow (2,6) \rightarrow (3,7) \rightarrow (4,8) \rightarrow \ldots$$

After S steps we hit the node $(3,7)$ where we started from, since macro- and micro-structure represent the same sequence. They are only shifted.

All the separated rings can be derived by starting with node $(3,X)$ and $X \in \{1,2,3,\ldots S\}$. Each staring node will be met again after approaching $S-1$ other nodes. Hence, we get S disjunctive sequences, that is, S separated rings, each of length S. This result matches with the number of S^2 edges in the resulting adjacency matrix.

With this understanding, we can derive the result if mask and graph are representing rings of different size (m and n). If m is divisor of n or vice versa then the result is a disconnected graph, otherwise the graph is connected. That means, as long a mask is also initial structure the result is always a unconnected graph.

However, if we want to create graphs beyond rings, we can guaranty connectedness for masks forming a ring by adding at least one self-connection. This can be explained by going back to the example given above, where mask and original graph have the same segmentation and forming a single ring. Further we assume, node 3 in the original mask has now an additional self-connection. This has the effect that the following sequence is present in the resulting graph too:

$$(3,7) \rightarrow (3,8) \rightarrow \ldots$$

$$\cdots \rightarrow (3,S-1) \rightarrow (3,S) \rightarrow (3,1) \rightarrow (3,2) \rightarrow \ldots$$

$$\cdots \rightarrow (3,6) \rightarrow (3,7) \rightarrow (3,8) \rightarrow \ldots$$

In the graph M_1 (i.e. having no self-connection) each of the nodes $(3,X)$, $1 \leq X \leq S$, is located on a different ring. Therefore, the additional ring generated by the self-connection builds a "junction"

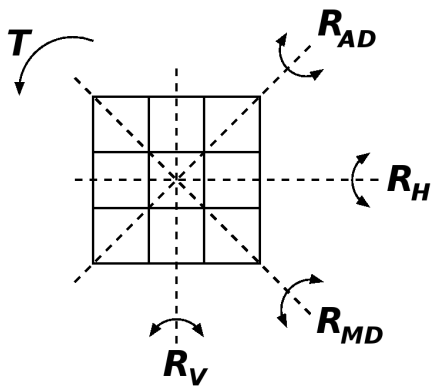


Figure 5. General rotation and reflection symmetries of Sierpiński carpets. Certain Sierpiński carpets remain the same applying a rotation (T) as well as under reflections along the middle horizontal or vertical line (R_H and R_V) as well as along the two diagonals R_{MD} and R_{AD}.

connecting all former separated rings. In consequence, each node can be reached from any other node. In other words the directed graph is connected.

In conclusion, we can summarize the general criteria for our generation process of connected graphs as follows: *each ring with an additional self-connection used as mask and initial graph generates a connected graph.*

2.3 Fractal dimensions of connected graphs

It is easy to see that a mask fully labeled generates only fully connected graphs. Hence, the non-trivial cases of connected graphs are generated by masks with n labeled entries, where $S < n < S^2$ and S is the segmentation of the mask. Interestingly enough, masks with this number of labeled segments generate Sierpiński carpets of fractal dimensions d_f between 1 and 2 [10], since:

$$d_f = \frac{log(n)}{log(S)},$$

from which follows:

$$1 < d_f < 2.$$

Therefore, we define a graph G as a *graph of fractal dimension* if G is connected and directed as well as the result of a mask of fractal dimension d_f, where $1 < d_f < 2$.

$\lvert E_0\rvert$	$d_f = \frac{\lvert E\rvert}{S}$	M_n^3	$\sum$
4	1.26	99, 102,106, 114	4
5	1.46	79, 94, 103, 107, 110, 115, 118, 122, 171, 173, 174, 186, 229, 355	14
6	1.63	95, 111, 119, 123, 126, 175, 187, 189, 190, 231, 238, 245, 335, 359, 363, 371, 427	17
7	1.77	127, 191, 239, 247, 254, 351, 367, 375, 379, 431, 443	11
8	1.89	255, 383, 447, 495	4
			$\sum = 50$

Table 1. Numbers of unique masks generating connected graphs and their fractal dimension.

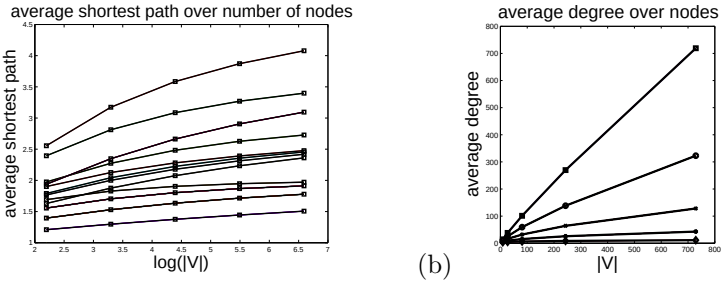
2.4 Symmetrical masks

Sierpiński carpets can be symmetric. Figure 5 shows the five possible transformations under which certain sets undergo no changes. The Sierpiński carpet shown in Figure 1 remains obviously the same for each of the five transformation.

If we apply these transformations to adjacency matrixes then only reflections along the two diagonals as well as two successively applied rotations (T^2) keep the graph connected. Under the consideration of these three symmetries we get 50 distinct masks of segmentation 3 of fractal dimension generating connected graphs. All other masks produce either no connected graph at all or can be transformed into one of the 50 sets by a combination of the operations T^2, R_{MD} and / or R_{AD}. Table 1 summarizes this result. A 4-segmentation gives us 6692 unique masks out of $2^{16} = 65536$ possibilities.

3. Properties of graphs with fractal dimensions

In the following we investigate some properties of the connected graphs constructed by the masks given in Table 1. For all masks we applied five iterations and therefore the final graphs have $3^6 = 729$ nodes. We tested connectedness for all graphs explicitly, because it turns out that the former formulated criterion does not cover all cases of mask generating connected graphs. All mask and the resulting graphs are connected.



(a) (b)

Figure 6. Diagrams showing the evolution over five iteration of the average shortest path length with respect to the number of nodes (a) and the mean of the in-degree k_i (equal to k_o) (b). Notice, overlapping curves appear as one.

3.1 Shortest paths

The first important feature of a connected graph is the average of the shortest paths and its evolution over the generation process. Self-connections are not included in this calculation of the shortest paths.

As it is indicated by the left diagram in Figure 6 the average of the shortest paths scales slower than *log* with respect to the number of nodes.

3.2 Average degree

A solid judgment about the shortest paths has to include the corresponding average degree. Figure 6 shows the evolution of the out-degree over the iterations for all graphs. Surprisingly enough, for all iterations the graphs have only one of five different average degrees. This effect goes back to the ratio of edges and nodes in the mask and can be explain as follows. The average degree $\overline{k}$ is actually determined by the ratio of edges and nodes in a graph:

$$\overline{k} = 2\frac{|E|}{|V|}.$$

For our graphs we know that:

$$|E_n| = E_0^n$$

and

$$|V_n| = S^n$$

where $|E_n|$ and $|V_n|$ is the number of edges and nodes after $n > 0$ iterations. Further on, E_0 is the number of edges in the mask and S is the segmentation of the applied mask, i.e. the number of nodes in

the corresponding graph. Hence, after n iterations we get the following average degree:

$$\overline{k}_n = 2 \cdot \left(\frac{E_0}{S}\right)^n .$$

The generation of connected graphs of fractal dimension means that E_0, a natural number, is between S and S^2 and therefore, it is limited. In consequence, we have a limited number of sets of graphs with equal average degree. In the 3-segmented case we have obviously five different average degrees. This number is determined by the possible values of E_0. For $S = 3$ we have five, since $E_0 \in \{4, 5, 6, 7, 8\}$), which is clearly indicated by the diagram in Figure 6(b).

A general properties of our fractal graphs is therefore, that *the average degree is exponentially increasing with respect to the number of nodes*, since per definition $\frac{E_0}{S} > 1$.

3.3 Clustering

The third characteristic of graph is the average clustering coefficient. Here, we applied the definition given by [16]. The resulting values over the iteration process for our graphs is plotted in Figure 7. One can see that all coefficient tend to decrease with respect to the size of the graph, i.e. $|V|$.

Clustering coefficients of graphs under investigations are often compared with the clustering values of random graphs $G_{n,p}$. A random graph belongs to the group $G_{n,p}$, if it is undirected and has n nodes, where each pair of nodes is connected with probability p [13]. In [16] the clustering coefficient of random graphs is given as:

$$c_{random} = \frac{\overline{k}}{|V|},$$

i.e. the ratio between average degree and the number of nodes in the graph. We have applied this definition of c_{random} in order to compare it with the clustering coefficients c_f of our fractal graphs, where

$$r = c_f \cdot \frac{|V|}{\overline{k}} = \frac{c_f}{2} \cdot \left(\frac{S^2}{E_0}\right)^n .$$

Considering $p = \frac{|E|}{2|V|^2}$ as the connection probability between two nodes in a random graph with equal number of nodes and edges, we get:

$$r = \frac{c_f}{p}.$$

The value p can be used as measure for the number of edges in our fractal graphs. In consequence, the less the fractal dimension the more

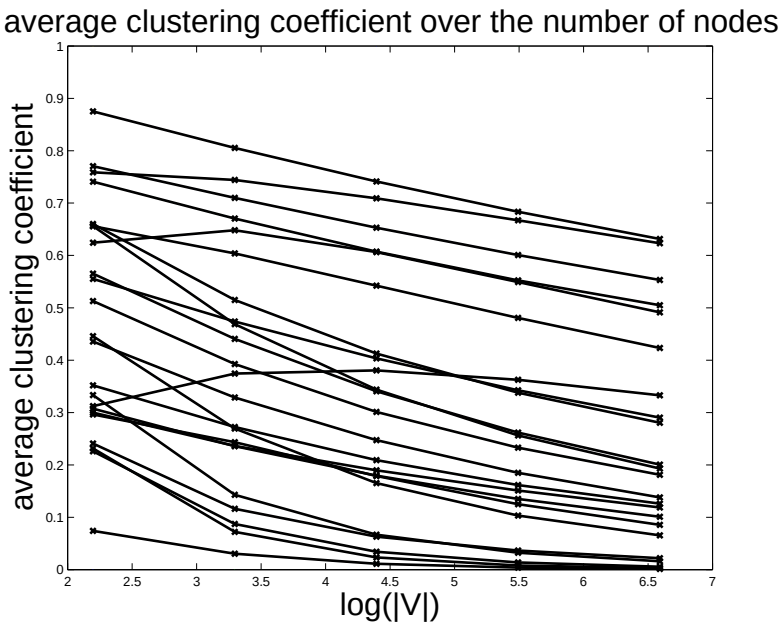


Figure 7. Clustering coefficient after the definition given in [16]

the cluster coefficient tend to be considerably higher compared with a random graph equal in the number of vertices and edges.

This is all we can say about the relation between clustering coefficients, since we have observed that c_f changes with respect to the number of nodes and we have no formal description for these changes. However, Figure 8 shows that the clustering of the fractal graphs can be considerably larger than for random graphs. As well as they can be much smaller. Again, it seems that the curves in 8 are establishing five distinct branches, which might be related to the five different fractal dimensions of the basic masks.

3.4 Degree distribution

The last property under investigation is the degree distribution. Similar to the average degree, this distribution is fully determined by the mask. Let:

$$d_i = k_o(v_i), 1 \leq i \leq S,$$

where S is the mask segmentation. Hence, d_i is the out-degree of node i. In the following we will only consider out-degrees. But the same argumentation can be applied for in-degree distribution.

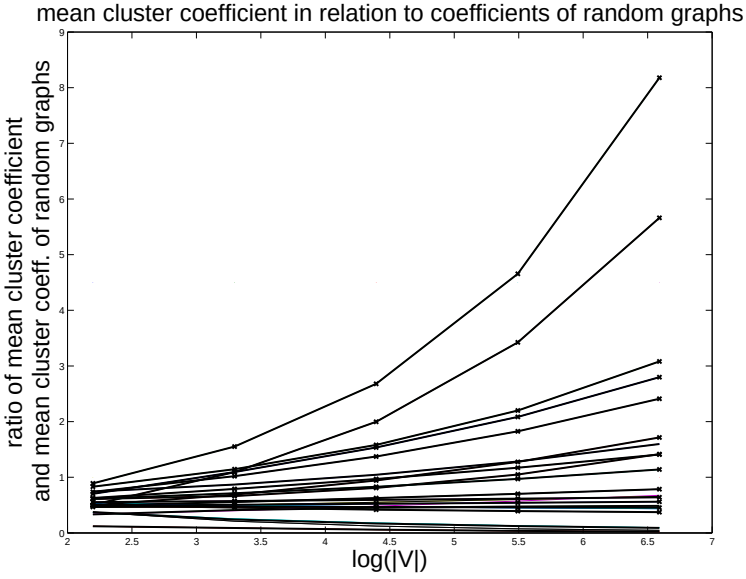


Figure 8. Relation between clustering coefficients of random graphs and our fractal graphs having the same size, i.e. equal number of edges and nodes.

The degree number of each node in a graph after j iteration can be derived by solving the equation:

$$(d_1 + d_2 + \cdots + d_S)^{j+1}, \quad j > 1,$$

in a symbolic manner. In that way we get a sum of S^{j+1} elements. Each summand represents the degree of a particular node and is a product of $(j + 1)$ factors. Each factor belongs to the set of degrees given in the originak mask. Hence, the degree distribution is directly given by the numerical values of these factors and the number of iterations.

As an example, assume we have:

$$d_i < d_S, \qquad 1 \leq i < S.$$

Thus after j iterations we have exactly one node, out of $n = S^{j+1}$ nodes, with the maximal out-degree ${d_S}^{j+1}$. Furthermore, let:

$$d_{1,\ldots,S-1} = 1$$

and

$$d_S > 1.$$

In such a case the number of nodes z_r with degree ${d_S}^{n-r}$ is given by:

$$z_r = \binom{n}{n-r} \cdot (S-1)^r, \quad 0 \leq r < n,$$

where S is the segmentation of the mask and $n = S^{j+1}$ is the number of nodes after j iterations. The term $\binom{n}{n-r}$ can be substitute by:

$$(n-r)^r \cdot \prod_{l=1}^{r} \left(\frac{1}{l} + 1\right)$$

and with n as the total number nodes, we determine the probability $P(r)$ for selecting a node with degree $d_S^{(n-r)}$ in the following way:

$$P(r) = \frac{z_r}{n} = \frac{(S-1)^r \cdot (n-r)^r}{n} \cdot \prod_{l=1}^{r} \left(\frac{1}{l} + 1\right).$$

By applying the following the estimation:

$$1 < \prod_{l=i}^{r} \left(\frac{1}{l} + 1\right) \leq 2^r$$

we get:

$$\frac{((S-1)(n-r))^r}{n} < P(r) \leq 2^r \frac{((S-1)(n-r))^r}{n}, \quad 0 \leq r < n.$$

It follows, that for this specific case, the tail of the corresponding degree distribution ($r > \frac{n}{2}$) is decreasing exponentially.

Obviously the degree distribution has no characteristic scale and due to the construction process there will be always "a few nodes" with a degree magnitudes larger then the average, called hubs [12]. In consequence we can say that *graphs of fractal dimension can represent scale-free networks.*

On the other hand we see, that in-degree and out-degree are totally independent. They are determined by the corresponding degrees given in the mask. Hence, the average degree as well as the distribution of the resulting graph is the sum of this quantities over the in- and out-degrees.

3.5 Five examples representing the five possible fractal dimensions

Figure 9 summarizes our analysis showing the qualities under investigation for five examples: the average shortest path, clustering coefficient and the ratio of this coefficient related to random graphs. Each mask has different fractal dimension. The lower diagrams in the figure show the degree distribution after 5 iterations.

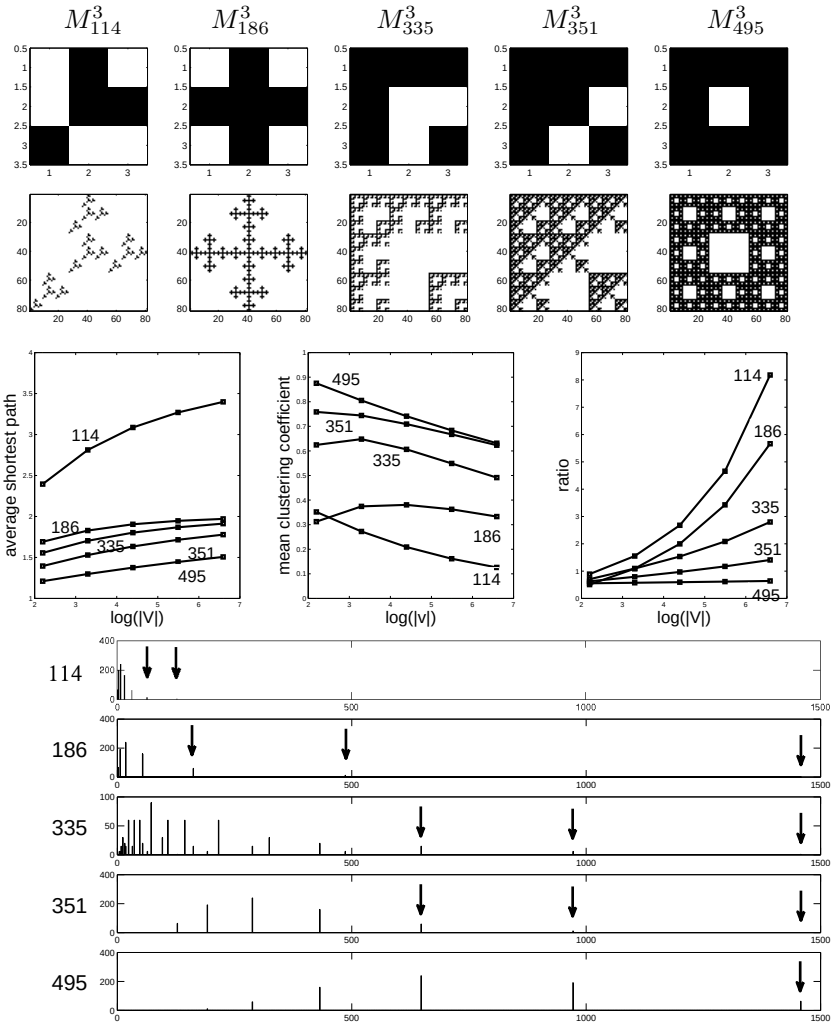


Figure 9. Characteristic qualities of five representative examples, incl. the degree distribution.

4. Robustness

The robustness of graphs can be investigated with respect to many specific properties. In the following we only consider the connectedness and the average length of the shortest paths while nodes are removed.

No matter which properties we are interested in, for any type of graph or network there are basically two kinds of robustness to consider. On the one hand, one wants to estimate how essential properties

of a network do change if a randomly chosen node is removed. On the other hand, it is important to know whether or not a graph contains single nodes which have radical impact on a global scale. Nodes of the latter type make a network highly sensitive to targeted attacks, since the failure of a single specific node causes a catastrophic failure of the whole network.

Due to the deterministic nature of our generation process, once more we can start to investigate these questions in a systematic manner.

4.1 Robustness under attack

Whether or not the connectedness of a given graphs is deteremined by a particular is node is verified by removing this node and testing connectedness explicitly While doing this for each node in the graph we get the number of nodes destroying connectedness if deleted.

All graphs during the first four iterations were analyzed in this way and Table 2 gives a summary of this experiment. One can see, that apart from 16 graphs all others have at least one node, which destroys the connectedness when removed. Interestingly enough, these cases can be distinguished with respect to the evolution of this number over the iterations. The number of this essential nodes is either exponetially increasing or remains constant.

We cannot explain yet, how this number of essential nodes is related to the underlying structure of the mask and the generation process. However, it turned out, that the number of single nodes, which can destroy connectedness is zero, if the mask has no node with in- and out-degree less than two, formally written as:

$$\forall i : k_o(v_i) > 1 \wedge k_i(v_i) > 1,$$

where v_i is a node in the graph represented by the mask.

The test of connectedness after the deletion of two nodes gives the same result (see Table 3). We only removed two nodes, if they alone can not destroy the connectedness. Hence, nodes counted in Table 2 are not considered in Table 3. Again, connectedness of a graph does not depend on only tow nodes, if the underlying mask contains no node which in- or out-degree less than 2.

This simple relation was tested for the first two iterations of the 4-segmented masks. It holds for these cases as well. Therefore, although not formally proven, our experiments give evidence, that *if a given mask of segmentation S represents a connected graph and each node in this mask has at least two out-going and two in-coming connections then the connectedness of the resulting graphs can not be destroyed by removing one or two nodes.*

E_0	M_n^3	1st	2nd	3rd	4th	E_0	M_n^3	1st	2nd	3rd	4th
4	99	7	15	31	63	6	231	1	1	1	1
	102	4	8	16	32		238	0	0	0	0
	106	4	8	16	32		245	0	0	0	0
	114	7	15	31	63		335	1	1	1	1
							359	2	2	2	2
5	79	1	1	1	1		363	1	1	1	1
	94	1	1	1	1		371	0	0	0	0
	103	4	8	16	32		427	0	0	0	0
	107	4	8	16	32						
	110	1	1	1	1	7	127	1	1	1	1
	115	2	2	2	2		191	1	1	1	1
	118	2	2	2	2		239	0	0	0	0
	122	4	8	16	32		247	0	0	0	0
	171	1	1	1	1		254	0	0	0	0
	173	2	2	2	2		351	0	0	0	0
	174	1	1	1	1		367	1	1	1	1
	186	1	1	1	1		375	0	0	0	0
	229	1	1	1	1		379	0	0	0	0
	355	2	2	2	2		431	0	0	0	0
							443	0	0	0	0
6	95	1	1	1	1						
	111	1	1	1	1	8	255	0	0	0	0
	119	1	1	1	1		383	0	0	0	0
	123	2	2	2	2		447	0	0	0	0
	126	1	1	1	1		495	0	0	0	0
	175	1	1	1	1						
	187	1	1	1	1						
	189	1	1	1	1						
	190	1	1	1	1						

Table 2. Numbers of nodes which turn the graph into a non-connected one, when deleted. This value is given for the first four iterations. Hence, the largest graphs have $3^5 = 243$ nodes.

4.2 Robustness under Failures

The experiments outlined in the previous paragraph indicate that the number of crucial nodes (if present) either remains constant or increases exponentially with respect to the iterations. Nevertheless, the node number is exponentially increasing as well and we get for the worst

E_0	M_n^3	1st	2nd	3rd	$E_0\|$	M_n^3	1st	2nd	3rd
4	99	0	0	0	6	231	3	3	4
	102	2	0	0		238	0	0	0
	106	2	0	0		245	0	0	0
	114	0	0	0		335	0	0	0
						359	0	0	0
5	79	0	0	0		363	2	3	4
	94	3	3	3		371	0	0	0
	103	0	0	0		427	0	0	0
	107	0	0	0					
	110	4	3	4	7	127	0	0	0
	115	4	6	8		191	0	0	0
	118	6	6	8		239	0	0	0
	122	0	0	0		247	0	0	0
	171	3	3	4		254	0	0	0
	173	6	6	8		351	0	0	0
	174	4	3	4		367	0	0	0
	186	0	0	0		375	0	0	0
	229	3	3	4		379	0	0	0
	355	4	6	8		431	0	0	0
						443	0	0	0
6	95	0	0	0					
	111	0	0	0	8	255	0	0	0
	119	2	3	4		383	0	0	0
	123	0	0	0		447	0	0	0
	126	3	3	4		495	0	0	0
	175	3	3	4					
	187	0	0	0					
	189	2	3	4					
	190	0	0	0					

Table 3. Numbers of pairs of nodes which turn the graph into a non-connected one, when deleted. A single node of these pairs does not destroy the connectedness.

cases (i.e. M_{99}^3 and M_{114}^3) the following relation:

$$p(n) = \frac{2^{(n+2)} - 1}{3^{n+1}},$$

where n stands for the n-th iteration. Hence, the probability of losing the connectedness in a graph of fractal dimension by accidently removing a single node is either zero or drops exponentially with respect to

the number of iterations.

Remains the question how other properties changes after a random deletion of one or more nodes without destroying the connectedness. We have investigated this question with respect to the average length of the shortest paths in a graph.

In these tests, the maximal value of shortest paths in a given graph is used for the normalization of the length of the shortest path. Hence, the normalized values indicate their length relative to the longest shortest path in a graph. Consequently, after removing one or more nodes the mean of these relative lengths is larger or equal compared with those of the original graph.

The three examples shown in Figure 10 are indicating the change of the shortest path lengths after removing up to 30 nodes. All removed nodes were selected in a way that the resulting graph was still connected. Two of these curves represent the very few examples where significant changes occur at all. Most of the graphs are represented by the curve of M^3_{126}, where values don't undergo significant alterations if nodes are deleted. This has several reasons. The first, of course, is simply the fact that the larger the fractal dimension the less impact has the deletion of nodes on a linear scale. On the other hand, nodes were randomly selected, but only removed, if connectedness wasn't destroyed. Therefore, most of the graphs were only cut on the periphery, where changes have minor impact on a global scale.

In summary we can say, *as long as connectedness in our graphs is not destroyed the deletion of a few nodes has no significant impact on the average length of the shortest paths.* As our experiments indicate, after removing up to 30 nodes in the given graphs, which originally consists of only 81 nodes, the length of the shortest path between most of the nodes remains the same as in the originial graph.

5. Discussion

In the two previous chapters we have intensively analyzed graphs generated with 3-segmented masks. Nevertheless, it was also shown, that many properties of the resulting complex structures can directly be related to the properties of the underlying masks. These insights provide us for determining and estimating graph properties resulting from higher segmentations. This section briefly summarizes these findings in order to outline a more general picture of the directed graphs of fractal dimension. Finally, an application for the generation of artificial neural networks is given.

5.1 Connectedness

An important issue of the here introduced method is that each iteration results in a connected graph. As we have shown above, as long as a

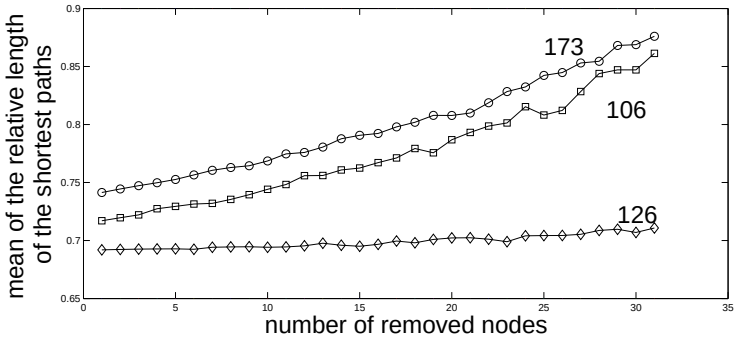


Figure 10. Evolution of the relative shortest path length in graphs resulting after 3 iterations applying mask M^3_{106}, M^3_{173}, and M^3_{126}. Up to 30 nodes were deleted in each graph. The relative length is related to the length of the longest shortest path in the original graph, which has 81 nodes. Each deletion process was repeated 50 times.

given mask is also applied as initial graph each iteration generates a connected graph, if the mask represents a ring plus one self-connection. We call these masks *complete*. Notice, this condition is sufficient for all higher segmentations as well.

In conclusion, each mask generates connected graphs, if it has a complete mask as sub-graph. Hence, it belongs to the set of complete masks.

However, which conditions guarantee connectedness in general, if an arbitrary mask M is applied to an arbitrary connected graph G? The argumentation above holds, if M is a complete mask and G is either a ring or the result of a complete mask. But for all other cases, this question, whether or not the result will be again a connected graph, remains open.

Nevertheless, as long as G and M are complete masks itself or are results of complete masks connectedness remains. In consequence, one can think about applications where different complete masks are used during the generation process. But this is an issue for future research or more concrete applications.

5.2 Parameterized scale-free graphs

As long the degrees in the mask have different values the resulting graphs show scale-free properties. (As an example for non-scale-free networks, one can think of masks where all nodes have the same in- and out-degree.)

The resulting average degree and degree distribution can be easily

derived from in- and out-degrees given in the mask and the number of iterations. Robustness issues can also directly addressed under the consideration of minimal in- and out-degree present in a given mask. As we saw, as soon as each in- and out-degree in the mask is larger one, the generated networks are robust against targeted attacks. This property holds for higher segmentations without any exception.

Considering these issues, one is able to construct networks with fractal dimension in a deterministic fashion which presents specific values of important network characteristics, such as average degree, degree distribution as well as robustness against failures and attacks.

5.3 Small-world properties

Small-world properties of networks are not uniformly defined in the literature [12, 13]. In consequence, it somehow depends on the definition applied, whether or not a graph establishes a small-world network. Furthermore, small-world properties are very often defined in relation to random graphs. This class, however, does not well match the deterministic nature of our fractal graphs. It is therefore not surprising, that graphs of fractal dimension are small-world networks as well as they are not. It depends on the given definition or, in other words, on the point of view.

A definition for small-world networks, given in [13], relates average degree and shortest paths, since a graph is a small-world network if the average length of the shortest paths scales *log* with respect to the number of nodes, while the average degree $\overline{k}$ is fixed.

As we have argued above, the average degree of our graphs increases exponentially and so this criterion is not fulfilled, since:

$$\overline{k} = \left(\frac{S+m}{S}\right)^n .$$

Again, S is the segmentation and n represents the iterations, while $S < (S+m) < S^2$. With respect to this relation, one can also see, that the average degree does not significantly change, if n and m are kept in certain ranges. For instance, set $S = 10$ and consider complete masks with less then 15 edges. The average degree before the first iteration is less 1.5. After 6 iterations we have a connected network of $n = 10^6$ nodes, while the average degree is still less than 5 and the mean of the shortest paths scales with $log(n)$. This example shows, that for specific cases, one might argue to have small-world properties, since $\overline{k}$ does not change significantly, while the number of nodes rises immensely.

Another characterization of small-world networks is based on the clustering coefficient. In random graphs the clustering coefficient tends to be $O(n^{-1})$ for large n (number of nodes), while small-world graphs are characterized by a $O(1)$-relation [13]. In other words, in small-world

networks the coefficient remains at least constant, which is obviously not the case for our graphs (see Figure 7). Again, the decrease of the clustering coefficient scales with *log*. For specific application this could mean, that the change is not significant. Does this mean, that under certain circumstances our graphs can be considered as examples of small-world networks?

A last characterization of small-world networks relates the clustering coefficient to random graphs. The clustering coefficient tends to be considerably higher than for a random graph of the same size, (i.e. equal number of nodes and edges) [16]. This relation is plotted in Figure 8 for $S = 3$. Again, there are graphs having clustering coefficients magnitudes larger than those of the corresponding random graphs. We also have argued above, that this relation correlates to the number of connections, which means, the less the fractal dimension the larger the coefficient in relation to random graphs. Furthermore, our diagrams give evidence that in some graphs the ratio linearly increases with respect to the number of nodes. In this sense, one could argue that connected graphs of small fractal dimension have small-world properties, because their clustering coefficient is much larger than for random graphs.

Unfortunately, the less the fractal dimension the less the absolute value of the clustering coefficient (see Figure 9). The classical Watts-Strogatz networks for generating small-world networks starts with a clustering coefficient of 0.5 [12, 13, 16]. Whether or not this value has to be considered as a threshold for networks with small-world properties is out of the scope of this paper.

In summary, our fractal graphs do not belong to the class of small-world networks in general. However, for specific parameter settings such network properties might be expected to emerge.

5.4 From Carpets to neural networks

Basically, there are two strategies for the implementation of ANN based on the introduced method (see Figure 11). First, the adjacency matrix / the graph can directly be interpreted as a neural network containing recurrences of any kind. Second, the adjacency matrix can purely be seen as the connections between two separated set of neurons of the same size. In this sense, a feed-forward structure between two neuron layers is created. Obviously, a chain of different feed-forward connections can be built in that way.

The former, the recurrent, case might be interesting as method for the generation of reservoirs of non-linear dynamics. Based on random graphs, this has been done in the echo-state [8] or liquid-state-machine [11] approach.

The latter case, might become an object of investigation within the

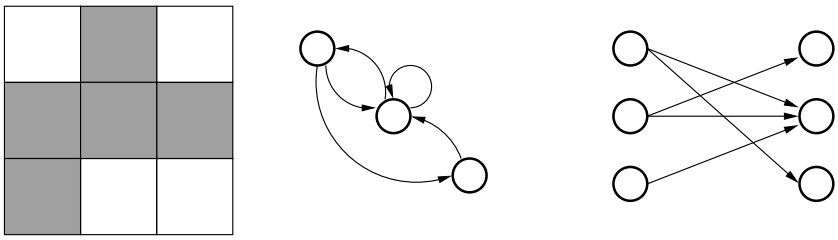


Figure 11. Two ways of transforming a given mask / graph (left) into a artificial neural networks. First, the graph is directly interpreted as ANN with recurrent neural connections (middle). Second, the adjacency matrix as a description of a feed-forward network (right).

Neural Darwinism approach to the function of the brain introduced, developed and promoted by Edelman [4]. According to this approach, an essential element for the brain-function is the matching between specific signal configurations and neural groups, which respond in a specific manner. Obviously, this matching must be sufficient specific in order to allow distinction among different signals, called recognition. However, more important within the Neural Darwinism approach is the argumentation, that such a matching must be degenerated. The assumption is, that there is more than one way to recognize a signal, that is, one signal configuration activates different neural groups as well as one neural group can be activated by different signal configurations. Two extremes of degeneration can be distinguished: a non-degenerated (unique) matching on one side and the completely degenerated matching on the other side. The Neural Darwinism approach claims that the variability of brain functions occurs within neural organization somehow located between these two extremes of non- and complete degeneration.

It is interesting to see, that the introduced graphs of fractal dimensions create networks between these to extremes. The examples shown in Figure 12 are only a simple schemas. But it is not hard to imagine that the fractal dimension and degree distribution of a graph determine the grad of degeneration. Therefore, we argue, that within the Neural Darwinism approach the introduced graphs of fractal dimensions might be a promising substrate for future research in order to model brain-like mechanisms of adaptation.

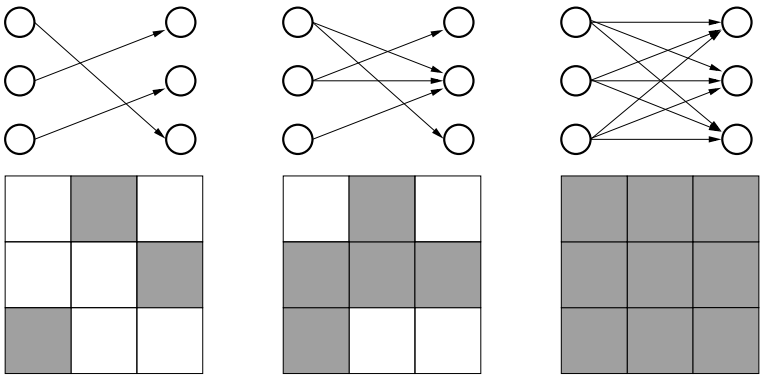


Figure 12. Three examples of feed-forward connections between two neural layers of an ANN. The left shows a non-degenerated matching between input and output signal. Each neuron in the left layer does only activate one neuron in the right layer. Due to the introduced representation this can be described as a ring. On the right a completely degenerated matching. Each neuron on the left activates each neuron on the right. This is represented by a fully connected graph. In between a degenerated matching formed by a graph of fractal dimension.

6. Conclusion

A method for the deterministic generation of directed and connected graphs was introduced. The average of the shortest paths in these graphs scale *log* with respect to the number of nodes. The generation process is inspired by fractal sets called Sierpiński carpets. We have shown that carpets of fractal dimensions between 1 and 2 can represent connected graphs. Therefore, we refer to them as connected graphs with fractal dimension.

We have formulated a general rule giving us a sufficient criterion supporting the evaluation whether or not a given mask produces connected graphs. For better or for worse, this criterion does not covers all possibilities of connected graph creating masks. At this point, we must be content to leave this problem to future investigations. However, the given criterion provides the construction of a huge number of masks and graphs; because the criterion is based on ring structures and each mask having such a structure as a sub-graph does generate connected graphs as well.

Due to the deterministic nature of the generation process we have shown that important properties of the resulting graphs can directly be derived from the structure of the mask used in the process. These properties are average degree and degree distribution. For specific degree distributions in the mask scale-free graphs emerge. We also have

discussed under which circumstances small-world networks be expected to emerge.

Further on, we have outlined conditions for the robustness of such networks against failures and attacks. These conditions are also valid for any segmentation larger 3.

The simplicity of the deterministic process support an applications of this method for many fields. As we have argued above, artificial neural networks might be a promising domain for further developments and future research.

References

[1] M. A. Arbib, *The metaphorical brain 2* (John Wiley & Sons, Inc., Canada, 1989).

[2] D. R. W. Barr, P. Dudek, J. Chambers and K. Gurney, "Implementation of Multi-layer Leaky Integrator Networks on a Cellular Processor Array" (in International Joint Conference on Neural Networks, IJCNN 2007, Orlando, Florida, 2007).

[3] B. Bollobás *Random Graphs* (Academic Press Inc., London, UK, 1985.)

[4] G. M. Edelman *Neural Darwinism* (Oxford University Press, Oxford, UK, 1989.)

[5] J. L. Elman, "Finding structure in time", (in Cognitive Science. 14:179–211, 1990.)

[6] J. Hertz, A. Krogh and R.G. Palmer, *Introduction to the theory of neural computation* (Perseus Books, USA, 1991).

[7] M. Ito, K. Noda, Y. Hoshino, and J. Tani, "Dynamic and interactive generation of object handling behaviors by a small humanoid robot using a dynamic neural network model", (in Neural Networks, 19:323–337, 2006).

[8] H. Jaeger, H. Haas, "Harnessing Nonlinearity: Predicting Chaotic Systems and Saving Energy in Wireless Communication", (in Science, 78-80, 2004).

[9] E. R. Kandel, J. H. Schwartz and T. M. Jessell, *Principles of Neural* (4th ed., McGraw-Hill Medical, New York, USA, 2000).

[10] B. B. Mandelbrot, *The Fractal Geometry of Nature* (Freeman, Scan Francisco, USA, 1982).

[11] W. Maass, T. Natschläger and H. Markram, "Real-Time Computing Without Stable States: A New Framework for Neural Computation Based on Perturbations", (in Computation, 14:2531-2560, 2002).

[12] S.N. Dorogovtsev and J.F.F. Mendes, *Evolution of Networks: From Biological Nets to the Internet and WWW* (Oxford University Press, Oxford, UK, 2003).

[13] M. Newman, "The structure and function of complex networks", in (M. E. J. Newman. The structure and function of complex networks. SIAM Review, 45(2):167–256, 2003).

[14] R. Pfeifer and C. Scheier, *Understanding Intelligence* (MIT Press, Cambridge, USA, 2000).

[15] D. E. Rumelhart and J. L. McClelland, *Parallel Distributed Processing* (MIT Press, USA, 1986).

[16] D. J. Watts, *Small Worlds: The Dynamics of Networks between Order and Randomness* (Princeton University Press, Chichester, UK, 1999).

[17] S. Wermter, G. Palm and M. Elshaw (Eds.), *Biomimetic Neural Learning for Intelligent Robots* (Springer, Heidelberg, Germany, 2005).